Quantum and classical parallelism in parity algorithms for ensemble quantum computers

Ralf Stadelhofer,[1,*] Dieter Suter,[2] and Wolfgang Banzhaf[3]
[1]*University of Dortmund Department of Computer Science, 44221 Dortmund, Germany*
[2]*University of Dortmund, Department of Physics, 44221 Dortmund, Germany*
[3]*Memorial University of Newfoundland, Department of Computer Science, St. John's, NL, A1B 3X5, Canada*
(Received 22 October 2004; published 28 March 2005)

The determination of the parity of a string of N binary digits is a well-known problem in classical as well as quantum information processing, which can be formulated as an oracle problem. It has been established that quantum algorithms require at least $N/2$ oracle calls. We present an algorithm that reaches this lower bound and is also optimal in terms of additional gate operations required. We discuss its application to pure and mixed states. Since it can be applied directly to thermal states, it does not suffer from signal loss associated with pseudo-pure-state preparation. For ensemble quantum computers, the number of oracle calls can be further reduced by a factor 2^k, with $k \in \{1,2,\ldots,\log_2(N/2)\}$, provided the signal-to-noise ratio is sufficiently high. This additional speed-up is linked to (classical) parallelism of the ensemble quantum computer. Experimental realizations are demonstrated on a liquid-state NMR quantum computer.

DOI: 10.1103/PhysRevA.71.032345 PACS number(s): 03.67.Lx

I. INTRODUCTION

Digital information processing relies on a number of error checking and correction algorithms. The most basic form of error detection checks the parity, which indicates if the number of 1's in a binary string is even or odd. The number of computational steps required to determine the parity of a binary string increases linearly with the length of the string; this holds true for classical as well as for quantum information processors [1]. Quantum algorithms can reduce the number of steps required by a factor of 2 compared to classical algorithms [1,2].

Apart from error correction, the parity problem has received significant attention in quantum information processing (see, e.g., Ref. [3]), since the parity of a product of binary strings can be used for efficiently searching a database [4,5].

For the analysis of the parity problem, it is often useful to formulate it as a black-box problem [1,6]. The black box, also referred to as an oracle, consists of N Boolean variables x_i: $X=(x_0,x_1,\ldots,x_{N-1})$, where $x_i \in \{0,1\}$. On input i, the oracle returns the Boolean variable x_i. Usually one wants to compute a property $p(X)$ of such a black box using as few oracle queries as possible. The number of these oracle calls is also called query complexity, which is the relevant complexity measure in this context; the total number of gates used is not considered.

When the parity problem is formulated as a black-box problem, the desired property is the parity, which can be written as the Boolean function

$$p(X) = x_0 \oplus x_1 \cdots \oplus x_{N-1}. \tag{1}$$

Here $\oplus$ denotes the XOR operation (addition mod 2). A classical computer has to call the oracle with each of the N possible inputs i to determine $p(X)$, while Beals *et al.* [1] and Farhi *et al.* [2] showed that in a quantum computer the minimum number of oracle calls is $N/2$.

In this paper, we discuss a quantum algorithm that is optimal in the sense of Refs. [1,2] and can be applied to pure as well as mixed states. Variants of this algorithm can be optimized for the application to ensemble quantum computers in such a way that the number of oracle calls decreases exponentially compared to single-issue quantum computers. The algorithm discussed here was developed using an automatic algorithm design technique called genetic programming.[1]

The paper is structured as follows: In Sec. II, we discuss the basic algorithm and apply it to a single-issue quantum computer. In Sec. III we apply it to an ensemble of quantum computers and in Sec. IV we show how a further reduction of the number of oracle calls is possible on ensemble quantum computers. Section V contains the experimental implementation on an NMR quantum computer and Sec. VI draws conclusions.

II. OPTIMAL EXACT QUANTUM ALGORITHM

The oracle gate whose parity we wish to calculate acts on N possible inputs i, which are encoded into $n \geqslant \log_2 N$ qubits. If N is not a power of 2, the string is extended with zeros. From now on we thus assume that $N=2^n$.

The gates used by the algorithm are the Hadamard operation **H**, the NOT operation σ_x, and the n-qubit oracle gate **O**. Upon input of a basis state $|i\rangle$, the oracle gate **O** returns the value of bit $x_i \in X$ in the form of a phase shift applied to the quantum register state:

$$\mathbf{O}|i\rangle = (-1)^{x_i}|i\rangle. \tag{2}$$

In the simplest case of a one-qubit quantum register ($N=2$, $n=1$), which is equivalent to Deutsch's problem [7], the

*Electronic address: ralf.stadelhofer@udo.edu

[1]For the use of genetic programming in evolving quantum algorithms see [13]. A general overview of genetic programming can be found in [14].

parity of the string can be determined with a single oracle call: With the qubit initialized in the $|0\rangle$ state, we apply an oracle gate bracketed by two Hadamard gates. The resulting state of the quantum register is

$$|\psi_{final}\rangle = \mathbf{HOH}|0\rangle = \mathbf{HO}\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$= \mathbf{H}\frac{1}{\sqrt{2}}[(-1)^{x_0}|0\rangle + (-1)^{x_1}|1\rangle]$$

$$= p(X)|1\rangle + [1 - p(X)]|0\rangle, \quad (3)$$

up to a global phase factor. Readout of the qubit shows the parity of X: even parity $[p(X)=0]$ results in a final state $|\psi_{final}\rangle = |0\rangle$, while odd parity results in $|\psi_{final}\rangle = |1\rangle$. The speed-up by a factor of 2, compared to the classical algorithm, results from the fact that the superposition determines whether the two bits are equal or opposite, but does not differentiate between, e.g., the strings "00" and "11."

To apply this algorithm to strings of arbitrary length N, we write the quantum register as

$$|\psi\rangle = |\chi\rangle \otimes |\xi\rangle, \quad (4)$$

where $|\chi\rangle$ contains the single qubit with index 0 that is used for readout, while $|\xi\rangle$ consists of the $n-1$ remaining qubits. All n qubits are first initialized into the $|0\rangle$ state; a Hadamard gate is then applied to the readout qubit to create the superposition state

$$|\psi_1\rangle = (\mathbf{H}|0\rangle) \otimes |0\cdots 0\rangle = \frac{1}{\sqrt{2}}(|00\cdots 0\rangle + |10\cdots 0\rangle). \quad (5)$$

If an oracle gate is applied to this state, it shifts the phase of each of the two components by π depending on the bit at position 0 or $N/2$ in X, respectively, being set. To take the other bits into account, we use repeated oracle calls with different inputs i. Since **O** does not modify the input vector $|\xi\rangle$, apart from the overall phase factor, we can generate the other inputs by subsequently flipping individual qubits. Figure 1 summarizes the resulting algorithm for $n=2$ and $n=3$ qubits. In the $n=2$ case, the $|\chi\rangle$ component subsequently takes the values 0 and 1, and in the $n=3$ case, it goes through $00\rightarrow 10\rightarrow 11\rightarrow 01\rightarrow 00$. The last step can be omitted, but will be assumed here for the convenience of making the final state independent of the sequence of single qubit flips. We summarize this sequence of N oracle calls alternating with σ_x equal to NOT operations with the unitary operator $\mathbf{U}_c$. Since its component operations **O** and σ_x are self-inverse and commute with each other, one gets $\mathbf{U}_c = \mathbf{U}_c^{-1} = \mathbf{U}_c^{+}$.

After this sequence of operations, the state of the quantum register is

$$|\psi_1\rangle = \mathbf{U}_c\mathbf{H}^{(0)}|00\cdots 0\rangle = \frac{1}{\sqrt{2}}[(-1)^{x_0\oplus x_1\cdots\oplus x_{N/2-1}}|00\cdots 0\rangle$$

$$+ (-1)^{x_{N/2}\cdots\oplus x_{N-1}}|10\cdots 0\rangle]. \quad (6)$$

The final Hadamard gate on the readout qubit transforms this state into

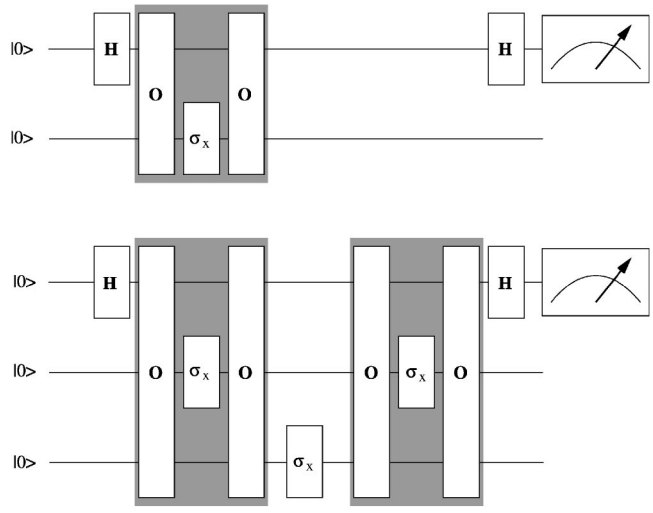

FIG. 1. Parity algorithm for $n=2$ qubits (top) and $n=3$ (bottom).

$$|\psi_{final}\rangle = \mathbf{H}^{(0)}|\psi_1\rangle = \begin{cases} |00\cdots 0\rangle & \text{if } p(X)=0, \\ |10\cdots 0\rangle & \text{if } p(X)=1. \end{cases} \quad (7)$$

The state of the readout qubit therefore codes the parity $p(X)$ of the string X.

The number of calls of the oracle gate $(N/2)$ required by this algorithm coincides with the lower bound established in Refs. [1,2]. Our algorithm is therefore optimal with respect to the number of oracle gates required, but also with respect to the number of additional gates, which are single-qubit gates, independent of the size of the quantum register. If any of the NOT gates were omitted, two oracle gates would become adjacent to each other. According to Eq. (2), the oracle is its own inverse, so they could be eliminated from the algorithm, thereby violating the lower bound. Our algorithm requires the measurement of a single qubit, in contrast to the $N/2$ measurements used by the algorithm proposed in [1] and to the n measurements required by the algorithm of Farhi *et al.* [2].

III. APPLICATION TO AN ENSEMBLE QUANTUM COMPUTER

We now turn to the discussion of ensemble quantum computers. To be able to discuss the operation of the algorithm on pure and mixed states within the same formal framework, we describe the state of the quantum register with a density operator. In most implementations, like in liquid-state NMR quantum computers, the initial state is the thermal state

$$\varrho_{th} \approx \frac{1}{N}(\mathbb{1} - \mathcal{H}) \approx \frac{1}{N}\left(\mathbb{1} - \sum_{i=0}^{n-1}\omega_i \mathbf{I}_z^{(i)}\right), \quad (8)$$

where we have set $\hbar/k_BT=1$ and invoked the high-temperature approximation. Here $\mathcal{H}$ denotes the Hamiltonian of the spin system, ω_i is the Larmor frequency of the ith spin (qubit), and $\mathbf{I}_z^{(i)}$ the corresponding spin operator.

The initial Hadamard gate on the readout qubit turns this state into

$$\varrho = \frac{1}{N}\left(\mathbb{1} - \omega_0 \mathbf{I}_x^{(0)} - \sum_{i=1}^{n-1} \omega_i \mathbf{I}_z^{(i)}\right). \tag{9}$$

The unity operator is time independent and does not contribute to any observable signal. The third term, which contains the thermal polarization of most of the spins, also does not contribute; we only need to consider the second term $\propto \mathbf{I}_x^{(0)}$. To compute the effect of the oracle gate on this term, we use the decomposition

$$\mathbf{I}_x^{(0)} = \frac{1}{2}\sum_{\xi=0}^{N/2-1} (|0\rangle\langle 1| \otimes |\underline{\xi}\rangle\langle\underline{\xi}| + \text{H.c.}), \tag{10}$$

where ξ stands for the binary representation of the natural numbers ξ.

The oracle gate turns this into

$$\mathbf{O}\mathbf{I}_x^{(0)}\mathbf{O} = \frac{1}{2}\sum_{\xi=0}^{N/2-1} (-1)^{x_{2\xi}\oplus x_{2\xi+1}}(|0\rangle\langle 1| \otimes |\underline{\xi}\rangle\langle\underline{\xi}| + \text{H.c.}). \tag{11}$$

Like in the single-instance case, we cycle the system through all possible oracle inputs by applying the sequence $\mathbf{U}_c$ of oracle gates and bit-flip operations σ_x. Each term in the above sum then acquires the same phase factor:

$$\mathbf{U}_c \mathbf{I}_x^{(0)} \mathbf{U}_c = \frac{1}{2}(-1)^{p(X)} \sum_{\xi=0}^{N/2-1} |0\rangle\langle 1| \otimes |\underline{\xi}\rangle\langle\underline{\xi}| + \text{H.c.} = (-1)^{p(X)} \mathbf{I}_x^{(0)}. \tag{12}$$

By measuring the sign of the resulting spin polarization of the readout qubit, we can therefore directly determine the parity of the string in a single measurement.

IV. SPEED-UP FOR ENSEMBLE QUANTUM COMPUTERS

While the determination of the parity requires at least $N/2$ oracle calls in a single-issue quantum computer, we now discuss a modified algorithm, which can determine the parity by a single oracle call, provided it runs on an ideal, noiseless ensemble quantum computer. This can be seen by calculating the expectation value of the observable $\mathbf{I}_x^{(0)}$ for the state (11) of the quantum register after the first call of the oracle gate:

$$\text{Tr}[\mathbf{I}_x^{(0)}\mathbf{O}\mathbf{I}_x^{(0)}\mathbf{O}] = \frac{1}{2}\sum_{\xi=0}^{N/2-1} (-1)^{x_{2\xi}\oplus x_{2\xi+1}}. \tag{13}$$

For even parity, the sum can reach extremal values of $\pm N/2$, and for odd parity, the extremal values are $\pm(N/2-2)$. The measured values are

$$\langle \mathbf{I}_x^{(0)}\rangle_{p(X)=0} = \frac{r\omega_0}{2N}; \quad r \in \left\{-\frac{N}{2}, -\frac{N}{2}+4, \ldots, \frac{N}{2}\right\},$$

$$\langle \mathbf{I}_x^{(0)}\rangle_{p(X)=1} = \frac{s\omega_0}{2N}; \quad s \in \left\{-\frac{N}{2}+2, \ldots, \frac{N}{2}-2\right\}. \tag{14}$$

A single call to the oracle gate thus allows one to determine the parity by measuring the expectation value of $\mathbf{I}_x^{(0)}$, provided the resolution of this measurement is high enough to distinguish between neighboring values.

This separation between neighboring values decreases with the length N of the string—i.e., exponentially with the number of qubits. The scheme is therefore not scalable for large systems, but even if the separation becomes too small to be resolved by the measurement, it remains possible to generate an exponential speed-up over the single-issue quantum computer at the cost of a correspondingly higher demand on the precision of the readout: The two cases that we have considered so far, using $N/2$ and a single oracle call, respectively, can be considered extreme cases of a series of algorithms that require 2^{n-k-1} calls of the oracle gate, corresponding to a speed-up by 2^k compared to the single-issue quantum computer.

For this purpose, we subdivide the address register (4) into three parts:

$$|\psi\rangle = |\chi\rangle \otimes |\mu\rangle \otimes |\nu\rangle, \tag{15}$$

where $|\chi\rangle$ is again the single readout qubit, while $|\xi\rangle = |\mu\rangle \otimes |\nu\rangle$ represents the remaining $n-1$ qubits. If the number of qubits in $|\nu\rangle$ is k, $|\mu\rangle$ contains only $n-k-1$ qubits.

We now restrict the number of oracle calls to all possible combinations of the qubits in $|\mu\rangle$—i.e., 2^{n-k-1}. The relevant term

$$\mathbf{I}_x^{(0)} = \frac{1}{2}\sum_{\mu=0}^{2^{n-k-1}-1}\sum_{\nu=0}^{2^k-1} (|0\underline{\mu\nu}\rangle\langle 1\underline{\mu\nu}| + \text{H.c.}) \tag{16}$$

in the density operator (9) is then transformed into

$$\mathbf{U}_c \mathbf{I}_x^{(0)} \mathbf{U}_c = \frac{1}{2}\sum_{\nu=0}^{2^k-1}\left[\left(\prod_{\mu=0}^{2^{n-k-1}-1} (-1)^{x_{0\underline{\mu\nu}}\oplus x_{1\underline{\mu\nu}}}\right) \times \sum_{\mu=0}^{2^{n-k-1}-1} (|0\underline{\mu\nu}\rangle\langle 1\underline{\mu\nu}| + \text{H.c.})\right], \tag{17}$$

where $\mathbf{U}_c = \mathbf{U}_c^{-1}$ represents the sequence of 2^{n-k-1} oracle and NOT gates.

Calculating the expectation value for this state, in analogy to Eq. (13), we find

$$\text{Tr}[\mathbf{I}_x^{(0)}\mathbf{U}_c\mathbf{I}_x^{(0)}\mathbf{U}_c] = 2^{n-k-2}\sum_{\nu=0}^{2^k-1}\left(\prod_{\mu=0}^{2^{n-k-1}-1} (-1)^{x_{0\underline{\mu\nu}}\oplus x_{1\underline{\mu\nu}}}\right). \tag{18}$$

Similar to the results from the single oracle call, the expectation value for $\mathbf{I}_x^{(0)}$ depends on the parity $p(X)$:

$$\langle \mathbf{I}_x^{(0)}\rangle_{p(X)=0} r\omega_0 2^{-k-2};$$

$$r \in \{-2^k, -2^k+4, \ldots, 2^k\},$$

$$\langle \mathbf{I}_x^{(0)}\rangle_{p(X)=1} = s\omega_0 2^{-k-2};$$

$$s \in \{-2^k+2, \ldots, 2^k-2\}. \tag{19}$$

Expectation values indicating opposite parities are thus separated by

$$|\langle \mathbf{I}_x^{(0)} \rangle_{p(X)=0} - \langle \mathbf{I}_x^{(0)} \rangle_{p(X)=1}| \geqslant \omega_0 2^{-k-1}. \quad (20)$$

The minimal separation therefore decreases exponentially with the number k of omitted address qubits or linearly with the number of oracle calls saved.

The algorithm proposed by Miao [8] shows a similar exponential decrease in the difference of the signal strength necessary to decide the parity problem. In contrast to Miao's approach we do not require nonunitary quantum operations. Since our algorithm works directly with the thermal mixed state, the signal strength suffers no exponential decrease if the number of qubits increases; this is similar to the modified Deutsch-Jozsa algorithm as proposed by Myers *et al.* [9].

While we have only discussed the application of this reduced algorithm to the thermal state in Eq. (9), it can equally be applied to pure states of the form

$$\varrho = \mathbf{H}^{\otimes n}|0\cdots 0\rangle\langle 0\cdots 0|\mathbf{H}^{\otimes n} = \frac{1}{N}\sum_{i,j=0}^{N-1} |i\rangle\langle j|. \quad (21)$$

V. EXPERIMENTAL IMPLEMENTATION

We implemented the two-qubit version ($n=2$, $N=4$) of the exact parity algorithm as well as the reduced ensemble algorithm with $k=1$ on a liquid state NMR quantum computer, using the spins of the ^{1}H and ^{13}C nuclei in a carbon-13 labeled chloroform molecule ($CHCl_3$) whose Hamiltonian is of the form ($\hbar=1$)

$$\mathcal{H} = -\omega_0^{(H)}\mathbf{I}_z^{(H)} - \omega_0^{(C)}\mathbf{I}_z^{(C)} + 2\pi J\mathbf{I}_z^{(H)}\mathbf{I}_z^{(C)}.$$

Here $\omega_0^{(H)}$ and $\omega_0^{(C)}$ denote the Larmor frequencies of the nuclear spins and J the strength of the scalar coupling between them. In the following, we will use a resonant rotating frame, where $\omega_0^{(H)}=\omega_0^{(C)}=0$. All experiments were performed at room temperature on a homebuilt NMR spectrometer with a ^{1}H operating frequency of 360 MHz.

The exact version of the parity algorithm, which needs two oracle calls, was implemented as shown in the upper part of Fig. 1. The first Hadamard gate **H** was replaced by the pseudo-Hadamard operation **h**, which corresponds to a $(\pi/2)_y$ rotation of the corresponding qubit around the y axis. The final Hadamard gate $\boldsymbol{h}^{-1}$ then cancels with the readout pulse that would otherwise be required to convert the state $|\psi_{final}\rangle$ [Eq. (7)] into observable $\mathbf{I}_x^{(0)}$ magnetization. The readout (of transverse magnetization) therefore starts immediately after the last oracle gate. As an additional simplification, we omitted the last bit reversal of the second qubit, which does not affect the readout qubit.

The $\boldsymbol{\sigma}_x$ operation (NOT gate) was realized by a $(\pi)_x$ pulse. The oracle gate **O** that represents the black box $X=(x_0,x_1,x_2,x_3)$ has the matrix representation

$$O = \begin{pmatrix} (-1)^{x_0} & & & \\ & (-1)^{x_1} & & \\ & & (-1)^{x_2} & \\ & & & (-1)^{x_3} \end{pmatrix}.$$

Thus the oracle gate for $X=(0,0,0,1)$ can be realized by the pulse sequence $\tau-(\pi/2)_{-z}^{C}-(\pi/2)_{-z}^{H}$, where $\tau=1/(2J)$ denotes the time of a free evolution period where the system evolves under the scalar spin-spin coupling, only. With $J=215$ Hz one gets $\tau=2.326$ ms. The $(\theta)_{\pm z}$ rotations cannot be implemented directly by radio frequency pulses and were thus realized by the composite pulse sandwich $(\pi/2)_x-(\theta)_{\pm y}-(\pi/2)_{-x}$ [10].

Similar sequences were determined for the other 15 oracle gates. The resulting oracle gates are pairwise equivalent, modulo an overall phase factor, for strings with inverted bit values. As an example, compare the matrix representations for $X=(0,0,0,0)$ and $X=(1,1,1,1)$, which correspond to $\pm\mathbb{1}$. Clearly, the overall phase factor does not affect the measured result. This ambiguity of the oracle gates is not critical for our application, since the corresponding string pairs always have the same parity.

The pseudopure state necessary for the pure-state algorithm was realized via temporal averaging [11]—i.e., by adding up the spectra of three experiments in which the populations of the states $|01\rangle$, $|10\rangle$, and $|11\rangle$ were cyclically permuted.

The free induction signals measured at the end of each parity algorithm were Fourier transformed and are displayed in Fig. 2 for all possible strings with $N=4$. The uppermost trace shows, as a reference, the reference spectrum obtained by applying a readout pulse directly to the thermal equilibrium state. The two resonance lines correspond to the two spin orientations of the second (^{1}H) spin, which are almost equally populated in thermal equilibrium. The other traces represent the Fourier-transformed free induction signals measured after applying the parity algorithm for the strings indicated to the pseudopure state $|00\rangle$. According to the theoretical result, we expect the sign of the ^{13}C signal to represent the parity of the string. This agrees with the experimental observation where the signal for the even-parity strings is positive while the signal for the odd-parity strings is negative.

In the pure-state algorithm, the second qubit is always in a definite state: $|0\rangle$ in the algorithm discussed in Sec. II, $|1\rangle$ if the final NOT operation is omitted. Accordingly, only one of the two ^{13}C resonance lines has a nonvanishing amplitude.

As discussed in Sec. III, the algorithm can also be applied to mixed states, thus eliminating the need to prepare a pseudopure state and avoiding the corresponding reduction of signal strength. We do not discuss the corresponding measurements here, but proceed directly to the reduced version where the number of oracle calls is reduced to one ($k=1$). Figure 3 shows the required sequence of gate operations.

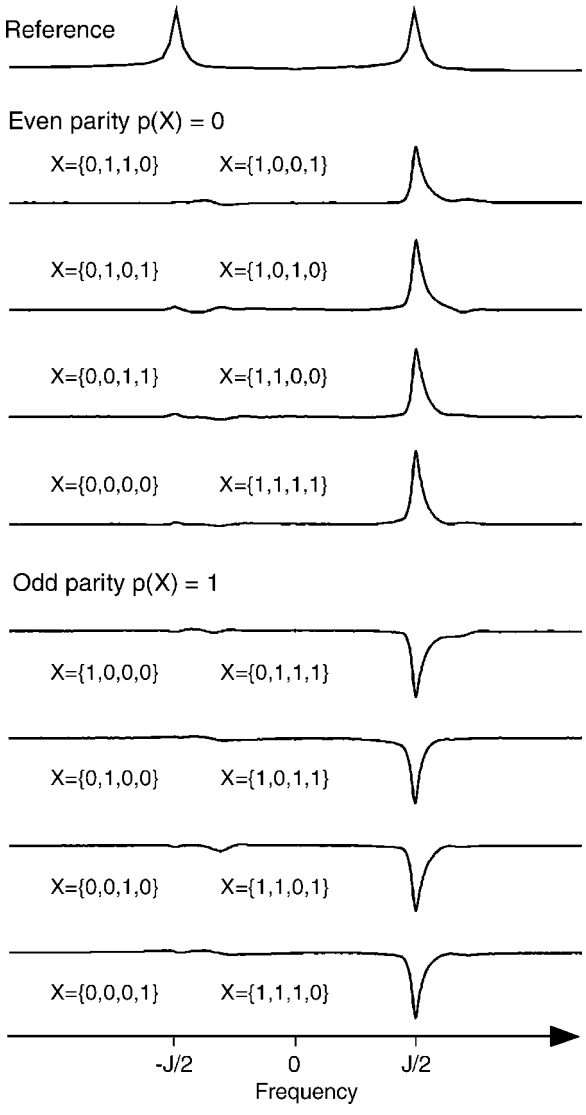

FIG. 2. Experimental results for the pure-state algorithm shown in Fig. 1. The uppermost trace shows the real part of the carbon spectrum after a readout pulse applied to the system in thermal equilibrium. The remaining spectra show the real part of the carbon spectrum after completion of the exact parity algorithm on the effectively pure initial state $|00\rangle$. The frequency is relative to 90.533 504 MHz.

Instead of the general Hadamard gate **H**, we again use the pseudo-Hadamard gate **h**—i.e., a $(\pi/2)_y$ pulse. The oracle gate is the same as in the pure-state case.

For the results of the reduced mixed-state algorithm, we only present the measurement results of $\langle \mathbf{I}_x^{(0)} \rangle$ at $t=0$—i.e., immediately after the end of the oracle gate. The results, shown in Table I, are $\pm(3.46\pm0.36)$ for the even-parity strings and $\pm(0.24\pm0.07)$ for odd-parity strings, in good agreement with the theoretical predictions of Eq. (14).

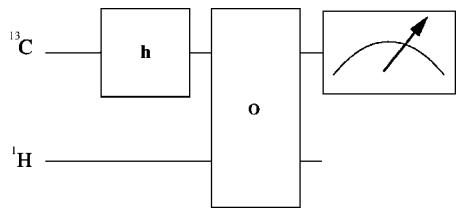

FIG. 3. Ensemble quantum algorithm for two qubits, using a single call of the oracle gate.

TABLE I. Experimental results of $\langle \mathbf{I}_x^{(0)} \rangle$ for the ensemble parity algorithm of Fig. 3. The numerical values are in arbitrary units.

$p(X)=0$	$\langle I_x^{(0)} \rangle$ $(t=0)$	$p(X)=1$	$\langle I_x^{(0)} \rangle$ $(t=0)$
$X=\{0,0,0,0\}$	3.84	$X=\{0,0,0,1\}$	−0.29
$X=\{0,0,1,1\}$	−3.29	$X=\{0,0,1,0\}$	0.18
$X=\{0,1,0,1\}$	3.77	$X=\{0,1,0,0\}$	−0.32
$X=\{0,1,1,0\}$	−2.95	$X=\{1,0,0,0\}$	0.16

VI. DISCUSSION AND CONCLUSION

We have introduced a family of quantum algorithms that solve the parity problem with an optimal number of quantum gate operations. It uses the black-box scheme introduced by Beals *et al.* [1] to represent the strings as oracle gates. In agreement with the lower bound established by Beals *et al.*, our algorithm uses $N/2$ calls of the oracle gate, a factor of 2 less than the best classical algorithms. This reduction compared to the classical case can be attributed to quantum parallelism, since the input state to the oracle gate is a superposition of two basis states.

A further reduction of the number of oracle calls is possible if an ensemble quantum computer is used rather than a single quantum system. In this case, the number of calls of the oracle gate can be reduced by a factor $2^k < N$, at the expense of a smaller separation between the measurement values that indicate even or odd parity. This additional speed-up requires parallel operation of many, nominally identical quantum systems, since a single system cannot provide the result in a single run.

The reduction of the number of oracle queries below the lower bound of $N/2$ may also be linked to the fact that ensemble quantum computers are able to distinguish nonorthogonal states, as mentioned by Dorai *et al.* in [12]. Such nonorthogonal states are the result of probabilistic quantum algorithms, such as the reduced parity algorithm introduced here.

The lower bound established in Refs. [1,2] was derived for exact as well as probabilistic algorithms. This is apparently in contradiction to our results for an ensemble of pure states, where the number of oracle calls $(=N2^{-k-1})$ can be arbitrarily smaller. The algorithm that we introduced here is, however, not covered by the usual discussion of probabilistic algorithms, where one assumes that the single-run error probability must be $<1/2$, while the error probability in our case is exactly $1/2$.

This work demonstrates again the usefulness and creativity of evolutionary methods for generating new algorithms.

ACKNOWLEDGMENTS

The authors thank Hans Georg Krojanski for technical support and helpful discussions as well as Dr. Xinuha Peng and Dr. Andre Leier for helpful discussions. Financial assistance was provided by the Deutsche Forschungsgesellschaft through Graduiertenkolleg 726.

[1] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf, J. ACM **48**, 778 (2001).
[2] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, Phys. Rev. Lett. **81**, 5442 (1998).
[3] E. Bernstein and U. Vazirani, SIAM J. Comput. **26**, 1411 (1997).
[4] B. M. Terhal and J. A. Smolin, Phys. Rev. A **58**, 1822 (1998).
[5] J. Du, M. Shi, X. Zhou, Y. Fan, B. Ye, R. Han, and J. Wu, Phys. Rev. A **64**, 042306 (2001).
[6] H. Burhman and R. de Wolf, Theor. Comput. Sci. **288**, 21 (2002).
[7] D. Deutsch, Proc. R. Soc. London, Ser. A **400**, 97 (1985).
[8] X. Miao, e-print quant-ph/0108116.
[9] J. M. Myers, A. F. Fahmy, S. J. Glaser, and R. Marx, Phys. Rev. A **63**, 032302 (2001).
[10] R. Freeman, T. A. Frenkel, and M. H. Levitt, J. Magn. Reson. (1969-1992) **44**, 409 (1981).
[11] E. Knill, I. Chuang, and R. Laflamme, Phys. Rev. A **57**, 3348 (1998).
[12] K. Dorai, Arvind, and A. Kumar, Phys. Rev. A **63**, 034101 (2001).
[13] L. Spector, H. Barnum, H. J. Bernstein, and N. Swamy, in *Advances in Genetic Programming* (MIT Press, Cambridge, MA, 1999), Vol. 3, p. 135.
[14] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone, *Genetic Programming: An Introduction* (Morgan Kaufmann, San Francisco, CA, 1998).